lendi Institute of Engineering & Technology
An Autonomous Institution
Accredited by NAAC with '' A'' Grade, Accredited by NBA (ECE, CSE.EEE & MECH)
Approved by A.I.C.T.E. & Permanently Affiliated to J. N. T. U. Gurajada, VIZIANAGARAM
Via 5th APSP Battalion, Jonnada (V), Denkada (M), NH-3, Vizianagaram Dist - 535005, A.P. Website : www.lendi.org
Ph : 08922-241111, 241666, Cell No : 9490344747, 9490304747,e-mail : lendi_2008@yahoo.com

---

PYTHON PROGRAMMING

II B. TECH I SEMESTER (R23)

STUDENTLABORATARY MANUAL

DEPARTMENT OF COMPUTER SCIENCE AND INFORMATION TECHNOLOGY

lendi Institute of Engineering & Technology
An Autonomous Institution
Accredited by NAAC with '' A'' Grade, Accredited by NBA (ECE, CSE.EEE & MECH)
Approved by A.I.C.T.E. & Permanently Affiliated to J. N. T. U. Gurajada, VIZIANAGARAM
Via 5th APSP Battalion, Jonnada (V), Denkada (M), NH-3, Vizianagaram Dist - 535005, A.P. Website : www.lendi.org
Ph : 08922-241111, 241666, Cell No : 9490344747, 9490304747,e-mail : lendi_2008@yahoo.com

DEPARTMENT OF COMPUTER SCIENCE AND INFORMATION TECHNOLOGY

VISION

- Producing globally competent and quality technocrats with human values for the holistic needs of industry and society

MISSION

- To impart strong theoretical and practical background in computer science and information technology discipline with an emphasis on software development.
- To provide an open environment to the students and faculty that promotes professional growth
- To inculcate the skills necessary to continue their education and research for contribution to society.

lendi Institute of Engineering & Technology
An Autonomous Institution
Accredited by NAAC with " A" Grade, Accredited by NBA (ECE, CSE.EEE & MECH)
Approved by A.I.C.T.E. & Permanently Affiliated to J. N. T. U. Gurajada, VIZIANAGARAM
Via 5th APSP Battalion, Jonnada (V), Denkada (M), NH-3, Vizianagaram Dist - 535005, A.P. Website : www.lendi.org
Ph : 08922-241111, 241666, Cell No : 9490344747, 9490304747,e-mail : lendi_2008@yahoo.com

----

Course Outcomes (CO's)

CO1: Understand the working environment of Python and its program structure.

CO2:Examine Python syntax and semantics and befluent in the use of Python flow control and functions.

CO3: Implement Conditionals and Loops for Python Programs

CO4: Use Python Lists, Tuples and Dictionaries for representing compound data.

CO5: Interpret the concepts of Object-Oriented Programming as used in Python

PROGRAM EDUCATIONAL OBJECTIVES (PEOs)

PEO1: Graduates will have strong knowledge and skills to comprehend latest tools and techniques of Computer Engineering so that they can analyze, design and create computing products and solutions for real life problems.

PEO2: Graduates shall have multidisciplinary approach, professional attitude and ethics, communication and teamwork skills, and an ability to relate and solve social issues through computer engineering.

PEO3: Graduates will engage in life-long learning and professional development to adapt to rapidly changing technology.

lendi Institute of Engineering & Technology
An Autonomous Institution
Accredited by NAAC with " A" Grade, Accredited by NBA (ECE, CSE.EEE & MECH)
Approved by A.I.C.T.E. & Permanently Affiliated to J. N. T. U. Gurajada, VIZIANAGARAM
Via 5th APSP Battalion, Jonnada (V), Denkada (M), NH-3, Vizianagaram Dist - 535005, A.P. Website : www.lendi.org
Ph : 08922-241111, 241666, Cell No : 9490344747, 9490304747,e-mail : lendi_2008@yahoo.com

---

PROGRAM OUTCOMES (POs)

PO1: Engineering Knowledge

PO2: Problem Analysis

PO3: Design & Development

PO4: Investigations

PO5: Modern Tools

PO6: Engineer & Society

PO7: Environment & Sustainability

PO8: Ethics

PO9: Individual & Team Work

PO10: Communication Skills

PO11: Project Mgt & Finance

PO12: Life Long Learning

PROGRAM SPECIFIC OUTCOMES (PSOs)

PSO1: Ability to solve contemporary issues utilizing skills.
PSO2: To acquire knowledge of the latest tools and technologies to provide technical solutions.
PSO3: To qualify in national and international competitive examinations for successful higher studies and employment.

lendi Institute of Engineering & Technology
An Autonomous Institution
Accredited by NAAC with '' A'' Grade, Accredited by NBA (ECE, CSE.EEE & MECH)
Approved by A.I.C.T.E. & Permanently Affiliated to J. N. T. U. Gurajada, VIZIANAGARAM
Via 5th APSP Battalion, Jonnada (V), Denkada (M), NH-3, Vizianagaram Dist - 535005, A.P. Website : www.lendi.org
Ph : 08922-241111, 241666, Cell No : 9490344747, 9490304747,e-mail : lendi_2008@yahoo.com

----

PYTHON PROGRAMMING SYLLABUS

Experiment1:
Programs on Basic I/O Sample

- **a.** Demonstrate the python script by running in Interactive and ScriptMode.
- **b.** Write a python script to read using input() and display using print() functions.
- **c.** Write a Python Program to Convert Celsius To Fahrenheit
- **d.** Write a Python program to compute area of triangle.
- **e.** Write a program to calculate the circumference of the circle
- **f.** WriteaPythonprogramtocomputedistancebetweentwopointsina2-dimensional Coordinate system.
- **g.** Write a program to swap two numbers without using a temporary variable.
- **h.** Write a Python program that calculates number of seconds in a day
- **i.** Write a python script to make use of all conversion functions.
- **j.** Demonstrate the following Operators in Python with suitable examples.
  i)Arithmetic Operators ii)Relational Operators iii)Assignment Operators
  iv) Logical Operators v) Bit wise Operators vi) Membership Operators
  vii) Identity Operator.

Experiment2:
Programs on Decision Making

- **a.** Write a program to find the largest element among three Numbers.
- **b.** Program to check whether a person is eligible to vote or not
- **c.** Write the python script to print whether the roots are equal, distinct or complex for given coefficients a, b and c for quadratic equation
- **d.** Write a Program to display all prime numbers within an interval
- **e.** Write a Python program to find the given year is leap year or not
- **f.** Write a program to find the factorial of a given number
- **g.** Write a python program to check whether given letter is vowels or not
- **h.** Write a python script to take five subject marks and print the grade for the student.
- **i.** Write a python script to read the person age from user and display Person status (1.age>60 senior citizen,2.age b/w 25 to 59 –working citizen,3.age b/w 16 to 24 college students ,4.age b/w 4 to 15 school kid,5.age b/w 1 to 3 play kid ,6.otherwise – invalid)
- **j.** Write a program to print the reverse of number.

Experiment3:

Programs on Loops(for,while)

- **a.** Write a program to calculate GCD of two numbers
- **b.** Write a Python program to print the multiplication table of a given number using a for loop. Take the number as input from the user.
- **c.** Write a Python program to print the first N Fibonacci numbers using a while loop. Take N as input from the user.
- **d.** write a program using for loop to calculate factorial of a number
- **e.** Write a Python program using while loop to print first N numbers divisible by 5
- **f.** Program to add natural numbers up to sum = 1+2+3+...+n
- **g.** Write a program to take input as integer N and check whether N is Pronic Number or not. (Product of two consecutive numbers is pronic N(N+1): Eg 110 = 10*11)
- **h.** Write a python script to take input as amount in rupees R and find out the least number of notes N that can be possible to store in a Wallet.(Hint Notes: 2000,500,200,100,50,20,10) Eg: R=2589, N=5
- **i.** Write a program to check whether given number N is N-Series(Disarium) number or not**.** (Eg. 135 is N-Series Number because $1^1+3^2+5^3$ = 135 and some others are 89, 175, 518 etc)

Experiment4:

Programs on Nested Loops(for,while)

a.Write a Python program to add two 3x3 matrices using nested loops. The matrices should be predefined

b.Write a python script to print the following pattern

```
1
1  2
1  2  3
1  2  3  4
1  2  3  4  5
```

c.Write a python script to print the following pattern

```
 1
 2  2
 3  3  3
 4  4  4  4
5  5   5   55
```

d.Write a python script to take input as String S="LENDI", print the following:

```
    L
   LEL
  LENEL
 LENDNEL
LENDIDNEL
```

e.Write a python script to print the any alphabet shape using *s.

```
  * * *
*       *
* *  *  *
*       *
*       *
```

Experiment5:

Programs on Modules & Functions

a. Write a program to define a function with multiple return values

b. Write a python script to implement different arguments in a function.

c. Write a program to define a function using default arguments

d. Write a python program to write the content “hi python programming” for the existing file?

e. Create a calculator module containing add, sub, mul and div and access them

f. Using Recursion, Write a program to take input as vehicle Number N and check whether N is Fancy number or not. (Folding of digits of number should be 9)

g.Create a module named “Lendi” and create functions addStudent, removeStudent, searchStudent. Access the above module using import statement.

h.Write a python script using lambdas, to take input as String, and sort the string SS in descending/ascending order according to their frequency of its occurrences of characters.(Eg. S='missisippi', SS=ispm)

i.Python program to check whether a JSON string contains complex object or not

Experiment6:

Programs Permutations &Combinations

a.Write a python script to take input as number N, and find out the largest number L , that can be formed with N.Eg. N=679, P={679,697,769,796,967,976}, L = 976

b.Write a python script to take input as list, L and print output as largest number L and total combinations C for given N digit number formed by the combination of L.(Eg. L=[1,2,1,4], N=3, L=421,C=12).

c.Write a python script to print Prime pairs within a given range of numbers. (Hint N=20, then (3,5) (5,7) (11,13) (17,19) are prime pairs)

d.By considering the terms in the Fibonacci sequence whose values do not exceed four million, find the sum of the even-valued terms.

Experiment 7:

Programs String &Regular Expressions

- **a.** Write a program to perform the given operations on a strings
- **b.** i) Creating the string ii) slicing the string iii) Delete character in the string
- **c.** Write a python script to take two string S1 and S2 and Check S1 and S2 are anagrams or not:
- **d.** Write a python script to take two string S1 and S2 and Check S1 is Sub string of S2 or not
- **e.** Write a python script to take two string S1 and S2 and check S1 is palindrome or not
- **f.** Write a Python program to reverse a given string using a for loop. Take the string as input from the user.
- **g.** Write a python script to take input as multi-line string and find the sum of all numbers in that string using re module. (Eg. S=”he11o they are 40students in97 room of 4th line”, Sum= 152)
- **h.** Using RegEx object check whether given phone number, email address and password is valid or not.
- **i.** Using date module, write a python script to take input as Date of birth (DOB) and current date(CD) and print age of the person.

Experiment 8:

Programs on Lists & Dictionary

a. Write a program to perform the given operations on a list:
i) Creating the list ii) slicing in the lists iii) Adding Elements in List
iv) Deleting the list elements

b. Write a program to count the number of vowels in a string(No control flow allowed).
c. Write a program to check if a given key exists in a dictionary or not.
d. Write a program to add a new key-value pair to an existing dictionary.
e. Write a program to take input as String S and print frequency of each character in S using List data structure.
f. Write a program to take input as String S contains characters and special symbols, reverse the String S such that special symbols remains at same position. (Eg. S=”m@d#u” , Output=”u@d#m”).
g. Write a python script to take input as String sentence S and print each word count using dictionary.
h. Write a python script to implement Anonymous function.
i. Write a python script to implement map(), reduce() and filter() functions

Experiment 9:

Programs on OOPS.

a. Write a Python program to create a person class. Include attributes like name, country and date of birth. Implement a method to determine the person's age.
b. Write a Python program to create a calculator class. Include methods for basic arithmetic operations.
c. Write a Python program to create a class representing a shopping cart. Include methods for adding and removing items, and calculating the total price.
d. Using Python OOPS, create a class, constructor, method,___str__ and___repr__ for
e. Employee
f. Using Python OOPS, create a class, constructor, method, ___str__ and___repr for
g. Student

Experiment 10:

Programs on Exceptions.

a. Write a python program to implement Exceptions hierarchy.
b. Write a program to Catching Specific Exceptions in Python
c. Python program to try with else clause.
d. Write a Python program to handle a ZeroDivisionError exception when dividing a number by zero.
e. Create a user defined Exception named “FundsLessException” and raise the exception when there are no enough funds in the bank account.

Experiment 11:

Programs Data Analysis

a. Python Program to demonstrate NumPy arrays creation using array()function.
b. Python script to load data sets.
c. Write a python script to create a data frame.
d. Python program to demonstrate use of ndim, shape,size,dtype.
e. Using NumPy, implement different matrix operations in python.
f. Using pandas, read the data from anytext files.
g. Python program to find min, max, sum, cumulative sum of array

h. Create a dictionary with at least five keys and each key represent value as a list where this list contains at least ten values and convert this dictionary as a pandas data frame and explore the data through the data frame as follows: Apply head() function to the pandas data frame

i. Perform various data selection operations on Data Frame

Experiment 12:
Programs on Plotting

a. Create a line plot for a list of values over a period of time. Label the x-axis as "Time" and the y-axis as "Value".
b. Create a bar chart that shows the frequency of categories in a dataset. Use different colors for each bar.
c. Create a histogram to visualize the distribution of a numerical dataset. Customize the number of bins and add a title.
d. Create a scatter plot to show the relationship between two numerical variables. Add a trend line to the scatter plot.
e. Create a pie chart to show the proportion of different categories in a dataset. Add labels and percentages to each slice.

COURSE OUTCOMES Vs PO's & PSO's

SNO	DESCRIPTION	PO(1..12) MAPPING	PSO(1..3) MAPPING
C210 3.1	UnderstandtheworkingenvironmentofPythonand itsprogramstructure.	PO1,PO2	PSO1,PSO2
C210 3.2	Examine Python syntax and semantics and be fluent in the use of Python flow control and functions	PO1,PO2,PO5	PSO1,PSO3
C210 3.3	Implement Conditionals and Loops for Python Programs.	PO1,PO2,PO3,PO4,PO5	PSO2,PSO3
C210 3.4	Use Python Lists, Tuples and Dictionaries for representing compound data.	PO1,PO2,PO3,PO4,PO5	PSO1,POS2, PSO3
C210 3.5	Interpret the concepts of Object-Oriented Programming as used in Python	PO1,PO2,PO4,PO5	PSO3
COURSE OVERALL PO/PSO MAPPING:			

COURSE OUTCOMES VS POs MAPPING (DETAILED: HIGH: 3, MEDIUM: 2, LOW: 1)

Course	SNO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
C2103	C2103.1	3	3	2	2	2	-	-	-	3	-	1	2	2	3
	C2103.2	3	3	2	2	3	-	-	-	3	-	1	3	3	3

	C2103.3	3	3	3	3	3	-	-	-	3	-	1	3	3	3
	C2103.4	3	3	3	3	3	-	-	-	3	-	1	3	3	3
	C2103.5	3	3	2	3	3	-	-	-	2	-	1	2	2	3
	C2103.*	3	3	3	3	3	-	-	-	3	-	1	3	3	3

**

**For Entire Course, PO & PSO Mapping*

Instructions to students

Pre-lab activities:

- Prepare observation note book which contains the following :
    - Procedure/algorithm/program to solve the problems discussed in the theory class
    - Solutions to the exercises given in the previous lab session
- Refer the topics covered in theory class

In-lab activities:

- Note down errors observed while executing program and remedy for that.
- Note down corrections made to the code during the lab session
- Answer to vivo-voce
- Get the observation corrected
- Note down inferences on the topic covered by the programs executed

Post-lab activities:

- Solve the given exercises
- Devise possible enhancements that can be made to the solved problem to simplify the logic
- Executed programs should be recorded in the lab record and corrected within one week after completion of the experiment.
- After completion of every module, a test will be conducted, and assessment results will have weight in the final internal marks.

General Instructions:

- Student should sign in the log register before accessing the system.
- Student is only responsible for any damage caused to the equipment in the laboratory during his session.
- Usage of pen drives is not allowed in the lab.
- If a problem is observed in any hardware equipment, please report to the lab staff immediately; do no attempt to fix the problem yourself.
- Systems must be shut down properly before leaving the lab.
- Please be considerate of those around you, especially in terms of noise level. While labs are a natural place for conversations regarding programming, kindly keep the volume turned down

WEEK 1

Experiment – 1:
AIM:: Study of Programs on BasicI/O Sample Programs

a.Demonstrate the python script by running in Interactive and ScriptMode.
b.Write a python script to read using input() and display using print() functions.
c.Write a Python Program to Convert Celsius To Fahrenheit

d.Write a Python program to compute area of triangle.

e.Write a program to calculate the circumference of the circle

f.WriteaPythonprogramtocomputedistancebetweentwopointsina2-dimensional Coordinate system.

g.Write a program to swap two numbers without using a temporary variable.

h.Write a Python program that calculates number of seconds in a day

i.Write a python script to make use of all conversion functions.

j.Demonstrate the following Operators in Python with suitable examples. i)Arithmetic Operators ii)Relational Operators iii)Assignment Operators iv) Logical Operators v) Bit wise Operators vi) Membership Operators vii) Identity Operator.

a)Demonstrate the Python Script by Running in Interactive and Script Mode

Description

Python code can be executed either interactively (line-by-line) or by running a script (a file with `.py` extension).

Algorithm

1. Launch Python shell for interactive mode.
2. Create and run a `.py` file using command line or IDE for script mode.

b)Script to Read Using input() and Display Using print()

Description

This program reads user input and displays it back using input() and print() functions.

Algorithm

1. Take input from user.
2. Print the input.

c)Convert Celsius to Fahrenheit

Description

This program converts temperature from Celsius to Fahrenheit using the formula:
$F = (C \times 9/5) + 32$

Algorithm

1. Read Celsius temperature.
2. Apply formula to convert to Fahrenheit.
3. Display result.

d)Compute Area of Triangle

Description

Calculates the area using formula:
$\text{Area} = \frac{1}{2} \times \text{base} \times \text{height}$

Algorithm

1. Read base and height.
2. Use formula to compute area.
3. Display result.

e)Circumference of a Circle

Description

Calculates the circumference using the formula:
$C = 2\pi r$

Algorithm

1. Read radius.
2. Apply formula.
3. Display result.

Distance Between Two Points in 2D

Description

Computes distance between (x1, y1) and (x2, y2) using:
$d = \sqrt{((x2-x1)^2 + (y2-y1)^2)}$

Algorithm

1. Read coordinates.
2. Compute difference and square.
3. Take square root.
4. Display distance.

g)Swap Two Numbers Without Temp Variable

Description

Uses Python tuple unpacking to swap two values without using a third variable.

Algorithm

1. Input two numbers.
2. Swap using a, b = b, a.
3. Print result.

h)Calculate Number of Seconds in a Day

Description

Computes total seconds in 24 hours:
seconds = 24 × 60 × 60

Algorithm

1. Multiply hours × minutes × seconds.
2. Print result.

i. Use of All Conversion Functions

Description

Demonstrates type conversion functions like int(), float(), str(), bool(), etc.

Algorithm

1. Take different data types.
2. Apply built-in conversion functions.
3. Display result.

j. Demonstrate Python Operators

Description

Shows usage of various operators: Arithmetic, Relational, Assignment, Logical, Bitwise, Membership, Identity.

Algorithm

1. Declare variables.
2. Apply each type of operator.
3. Display result.

VIVA QUESTIONS

- **What is the difference between interactive and script mode in Python?**
- **What does `input()` return in Python?**
- **How do you convert a string input to an integer?**
- **What is the formula to convert Celsius to Fahrenheit?**
- **What is the formula for the area of a triangle?**
- **What is the formula for the circumference of a circle?**
- **Which module provides π in Python?**
- **What is the distance formula in 2D?**
- **How can you swap two variables without a temporary variable in Python?**
- **How many seconds are there in one day?**
- **What does `bool(0)` return in Python?**
- **What is the difference between `list()` and `set()`?**
- **What is the difference between `is` and == in Python?**
- **What are the logical operators in Python?**

WEEK 2

Experiment2:

Aim: Programs on Decision Making

- **a.** Write a program to find the largest element among three Numbers.
- **b.** Program to check whether a person is eligible to vote or not
- **c.** Write the python script to print whether the roots are equal, distinct or complex for given coefficients a, b and c for quadratic equation
- **d.** Write a Program to display all prime numbers within an interval
- **e.** Write a Python program to find the given year is leap year or not
- **f.** Write a program to find the factorial of a given number
- **g.** Write a python program to check whether given letter is vowels or not
- **h.** Write a python script to take five subject marks and print the grade for the student.
- **i.** Write a python script to read the person age from user and display Person status (1.age>60 senior citizen,2.age b/w 25 to 59 –working citizen,3.age b/w 16 to 24 college students ,4.age b/w 4 to 15 school kid,5.age b/w 1 to 3 play kid ,6.otherwise – invalid)

a. Program to Find the Largest Among Three Numbers

Description:
Compares three numbers and prints the largest.

Algorithm:

1. Read three numbers.
2. Compare using if-elif-else.
3. Display the largest number.

b. Check Whether a Person is Eligible to Vote

Description:
Checks if a person is 18 years or older.

Algorithm:

1. Read age.
2. If age ≥ 18, print eligible. Else, not eligible.

---

c. Roots of a Quadratic Equation

Description:
Determines nature of roots using discriminant.

Algorithm:

1. Read a, b, c.
2. Calculate discriminant $d = b^2 - 4ac$.
3. Based on d, print root type.

d. Display All Prime Numbers Within an Interval

Description:
Prints all prime numbers in a given range.

Algorithm:

1. Input range.
2. For each number, check prime condition.
3. Print if prime.

e. Check if a Year is a Leap Year

Description:
Checks leap year using rules of divisibility.

Algorithm:

1. Input year.
2. Apply leap year conditions.
3. Display result.

f. Find Factorial of a Given Number

Description:
Calculates factorial using loop.

Algorithm:

1. Read number n.
2. Initialize fact = 1.
3. Multiply from 1 to n.
4. Print result.

g. Check Whether Given Letter is a Vowel

Description:
Checks if a letter is in vowel list.

Algorithm:

1. Input character.

2. Check if in a, e, i, o, u.
3. Print result.

---

h. Print Grade Based on Five Subjects

Description:
Takes marks of 5 subjects, calculates average, and prints grade.

Algorithm:

1. Read 5 subject marks.
2. Compute average.
3. Apply grading logic.

---

i. Display Person Status Based on Age

Description:
Categorizes person based on age brackets.

Algorithm:

1. Read age.
2. Use if-elif to check range.
3. Print status.

---

j. Reverse a Number

Description:
Reverses the digits of an integer.

Algorithm:

1. Input number.
2. Use loop to reverse.
3. Display reversed number.

Viva Questions

- **What is the difference between = and == in Python?**

- **What is the role of indentation in Python?**

- **How does Python handle input from users?**
- **How can you convert a string input to a number in Python?**
- **What is the purpose of loops in programs like factorial or prime checking?**
- **How do you handle multiple conditions in Python?**

WEEK 3

Experiment3:

Aim: Programs on Loops(for,while)

- **a.** Write a program to calculate GCD of two numbers
- **b.** Write a Python program to print the multiplication table of a given number using a for loop. Take the number as input from the user.
- **c.** Write a Python program to print the first N Fibonacci numbers using a while loop. Take N as input from the user.
- **d.** write a program using for loop to calculate factorial of a number
- **e.** Write a Python program using while loop to print first N numbers divisible by 5
- **f.** Program to add natural numbers up to sum = 1+2+3+...+n
- **g.** Write a program to take input as integer N and check whether N is Pronic Number or not. (Product of two consecutive numbers is pronic N(N+1): Eg 110 = 10*11)
- **h.** Write a python script to take input as amount in rupees R and find out the least number of notes N that can be possible to store in a Wallet.(Hint Notes: 2000,500,200,100,50,20,10) Eg: R=2589, N=5
- **i.** Write a program to check whether given number N is N-Series(Disarium) number or not**.** (Eg. 135 is N-Series Number because $1^1+3^2+5^3$ = 135 and some others are 89, 175, 518 etc)

a. Program to Calculate GCD of Two Numbers

Description:
This program calculates the greatest common divisor (GCD) of two numbers using the Euclidean algorithm.

Algorithm:

1. Read two numbers.
2. Use the Euclidean method to calculate the GCD:
   - While b is not zero, update a to b and b to a % b.
3. Print the GCD.

---

b. Program to Print the Multiplication Table of a Given Number

Description:
This program prints the multiplication table of a number entered by the user using a for loop.

Algorithm:

1. Read the number.
2. Use a for loop to multiply the number from 1 to 10.
3. Print the result.

---

c. Program to Print the First N Fibonacci Numbers Using a While Loop

Description:
This program prints the first N Fibonacci numbers using a while loop.

Algorithm:

1. Read the value of N.
2. Initialize the first two Fibonacci numbers, 0 and 1.
3. Use a while loop to print the next Fibonacci numbers until N numbers are printed.

---

d. Program to Calculate Factorial of a Number Using For Loop

Description:
This program calculates the factorial of a number using a for loop.

Algorithm:

1. Read the number.
2. Initialize a variable to store the factorial (starting with 1).
3. Multiply from 1 to the given number using a for loop.
4. Print the result.

---

e. Program to Print First N Numbers Divisible by 5 Using While Loop

Description:
This program prints the first N numbers divisible by 5.

Algorithm:

1. Read the value of N.
2. Use a while loop to check if a number is divisible by 5.
3. Print the number if divisible and count until N numbers are found.

---

f. Program to Add Natural Numbers Up to N

Description:
This program calculates the sum of natural numbers from 1 to N.

Algorithm:

1. Read the value of N.
2. Use the formula sum = n * (n + 1) / 2 or a loop to compute the sum.
3. Print the result.

---

g. Program to Check if N is a Pronic Number

Description:
This program checks if a number is pronic (product of two consecutive numbers).

Algorithm:

1. Read the number N.
2. Find the two consecutive numbers whose product equals N.
3. If such numbers exist, print that N is a pronic number.

---

h. Program to Find Least Number of Notes for a Given Amount

Description:
This program finds the least number of currency notes for a given amount in rupees.

Algorithm:

1. Read the amount.
2. Use division to determine how many notes of each denomination are needed.
3. Print the total number of notes.

---

i. Program to Check if N is a Disarium Number

Description:
This program checks if a number is a Disarium (N-Series) number.

Algorithm:

1. Read the number N.
2. Split the number into digits.
3. Calculate the sum of digits raised to the power of their position.
4. If the sum equals the original number, it is a Disarium number.

VIVA QUESTIONS

•What is the difference between a `while` loop and a `for` loop in Python?
•How do you calculate the factorial of a number in Python?.
•What is a Pronic number? How can you check if a number is Pronic?
•How does the `divmod()` function work in Python?
•What is a Disarium number, and how can you verify if a number is Disarium?
•How can you find the greatest common divisor (GCD) of two numbers in Python?

WEEK 4

Experiment4:

Aim: Programs on Nested Loops(for,while)

a.Write a Python program to add two 3x3 matrices using nested loops. The matrices should be predefined

b.Write a python script to print the following pattern

```
1
1  2
1  2  3
1  2  3  4
1  2  3  4  5
```

c.Write a python script to print the following pattern

```
 1
 2  2
 3  3  3
 4  4  4  4
5  5   5   5 5
```

d.Write a python script to take input as String S="LENDI", print the following:

```
    L
   LEL
  LENEL
 LENDNEL
LENDIDNEL
```

e.Write a python script to print the any alphabet shape using *s.

```
  * * *
*       *
* *  *  *
*       *
```

a. Python Program to Add Two 3x3 Matrices Using Nested Loops

Description:

This program adds two predefined 3x3 matrices using nested loops.

Algorithm:

1. Define two 3x3 matrices.
2. Initialize an empty result matrix of the same size.
3. Use nested loops to iterate through each element and sum corresponding elements of the matrices.
4. Print the result matrix.

b. Python Script to Print a Number Pattern

Description:
This script prints a pattern of numbers where each row contains numbers from 1 up to the current row number.

Algorithm:

1. Loop through numbers from 1 to 5.
2. In each iteration, print numbers from 1 to the current row number.

---

c. Python Script to Print Another Number Pattern

Description:
This script prints a pattern where the number is repeated on each row equal to the row number.

Algorithm:

1. Loop through numbers from 1 to 5.
2. In each iteration, print the current row number repeated the same number of times.

---

d. Python Script to Print a Pattern with String "LENDI"

Description:
This script prints a pattern based on the string "LENDI", where characters are added symmetrically.

Algorithm:

1. Initialize the string S = "LENDI".
2. For each iteration, print a part of the string from the beginning, and the same part reversed except for the last character.

---

e. Python Script to Print Alphabet Shape Using '*'

Description:
This script prints a shape resembling the letter "A" (or any other alphabet) using * symbols.

Algorithm:

1. Use loops to print spaces and stars in each row.
2. Align stars to form a specific alphabet shape.

Viva Questions:

- **What is the use of nested loops in matrix addition?.**
- **How do you print a pattern using loops in Python?**
- **Explain string slicing in Python with an example.**
- **Why is the end=" " used in the print() function?.**
- **How do you align stars to print a specific alphabet shape?**
- **What is the significance of range() function in loops?**

WEEK 5

Experiment5:

Aim: Programs on Modules & Functions

a. Write a program to define a function with multiple return values
b. Write a python script to implement different arguments in a function.
c. Write a program to define a function using default arguments
d. Write a python program to write the content "hi python programming" for the existing file?
e. Create a calculator module containing add, sub, mul and div and access them
f. Using Recursion, Write a program to take input as vehicle Number N and check whether N is Fancy number or not. (Folding of digits of number should be 9)
g.Create a module named "Lendi" and create functions addStudent, removeStudent, searchStudent. Access the above module using import statement.
h.Write a python script using lambdas, to take input as String, and sort the string SS in descending/ascending order according to their frequency of its occurrences of characters.(Eg. S='missisippi', SS=ispm)
i.Python program to check whether a JSON string contains complex object or not

a. Program to Define a Function with Multiple Return Values

Description:
This program defines a function that returns multiple values using a tuple.

Algorithm:

1. Define a function that returns multiple values.
2. Call the function and store the returned values.
3. Print the returned values.

---

b. Python Script to Implement Different Arguments in a Function

Description:
This program demonstrates how to pass different types of arguments to a function: positional, keyword, and arbitrary.

Algorithm:

1. Define a function with different types of arguments.
2. Call the function using different types of arguments.

---

c. Program to Define a Function Using Default Arguments

Description:
This program defines a function with a default argument, which can be overridden by the user.

Algorithm:

1. Define a function with a default argument.
2. Call the function without or with the argument.

---

d. Program to Write Content to an Existing File

Description:
This program opens an existing file and writes "hi python programming" to it.

Algorithm:

1. Open the file in write mode.
2. Write the content to the file.
3. Close the file after writing.

---

e. Create a Calculator Module and Access It

Description:
This program creates a calculator module with functions for addition, subtraction, multiplication, and division.

Algorithm:

1. Create a module with functions for basic operations.
2. Import and use the module in the main program.

---

f. Using Recursion, Check if a Vehicle Number is a Fancy Number

Description:
This program uses recursion to check if a number is a fancy number (the sum of digits must fold to 9).

Algorithm:

1. Define a recursive function to calculate the sum of digits.
2. If the sum is greater than 9, continue the process until the sum becomes a single digit.
3. Check if the final sum equals 9.

g. Create a Module Named “Lendi” with Functions for Student Operations

Description:
This program demonstrates creating a module to handle student-related operations like adding, removing, and searching students.

Algorithm:

1. Create a module Lendi with functions to add, remove, and search students.
2. Use the module in the main program.

h. Python Script Using Lambdas to Sort a String Based on Character Frequency

Description:
This script sorts a string based on the frequency of characters, either in ascending or descending order.

Algorithm:

1. Use a lambda function to calculate the frequency of each character.
2. Sort the string based on character frequency.

i. Python Program to Check if a JSON String Contains Complex Object

Description:
This program checks if a given JSON string contains any complex objects (like lists or dictionaries).

Algorithm:

1. Parse the JSON string into a Python object.
2. Check if the parsed object contains complex types.

Viva Questions:

1. **What is the difference between positional and keyword arguments?**
2. **How does recursion work in Python?**
3. **What is the purpose of the lambda function in Python?**
4. **How do you handle file operations in Python?**
5. **What is the role of modules in Python?**

WEEK 6

Experiment6:
Programs Permutations & Combinations
a.Write a python script to take input as number N, and find out the largest number L , that can be formed with N.Eg. N=679, P={679,697,769,796,967,976}, L = 976
b.Write a python script to take input as list, L and print output as largest number L and total combinations C for given N digit number formed by the combination of L.(Eg. L=[1,2,1,4], N=3, L=421,C=12).
c.Write a python script to print Prime pairs within a given range of numbers. (Hint N=20, then (3,5) (5,7) (11,13) (17,19) are prime pairs)
d.By considering the terms in the Fibonacci sequence whose values do not exceed four million, find the sum of the even-valued terms.

a. Python Script to Find the Largest Number from Permutations

Description:
This program finds the largest number that can be formed from the digits of a given number by generating all possible permutations.

Algorithm:

1. Take input as a number.
2. Generate all permutations of the digits of the number.
3. Convert the permutations to integers and find the largest one.

---

b. Python Script to Find Largest Number and Total Combinations

Description:
This program calculates the largest number that can be formed from a list of digits and counts the total number of distinct permutations that can be formed with the given number of digits.

Algorithm:

1. Take a list of digits and a number N.
2. Find all distinct permutations of N digits from the list.
3. Return the largest number and the total count of distinct permutations.

---

c. Python Script to Print Prime Pairs Within a Given Range

Description:
This program prints pairs of consecutive prime numbers within a given range.

Algorithm:

1. Generate a list of primes up to N.

2. Find pairs of consecutive primes.
3. Print the prime pairs.

---

d. Sum of Even-Valued Fibonacci Terms Not Exceeding Four Million

Description:
This program computes the sum of the even-valued terms in the Fibonacci sequence whose values do not exceed four million.

Algorithm:

1. Generate Fibonacci numbers until the value exceeds four million.
2. Sum the even Fibonacci numbers.

Viva Questions:

- **How do you generate permutations in Python?**.
- **What is the time complexity of generating all permutations of a number's digits?**
- **How do you check if a number is prime in Python**
- **Why do we use the set() function when generating permutations?**.
- **What is the Fibonacci sequence and how is it computed in this program?**

WEEK 7

Experiment 7:
Programs on Strings & Regular Expressions

- **a.** Write a program to perform the given operations on a strings
- **b.** i) Creating the string ii) slicing the string iii) Delete character in the string
- **c.** Write a python script to take two string S1 and S2 and Check S1 and S2 are anagrams or not:
- **d.** Write a python script to take two string S1 and S2 and Check S1 is Sub string of S2 or not
- **e.** Write a python script to take two string S1 and S2 and check S1 is palindrome or not
- **f.** Write a Python program to reverse a given string using a for loop. Take the string as input from the user.
- **g.** Write a python script to take input as multi-line string and find the sum of all numbers in that string using re module. (Eg. S=”he11o they are 40students in97 room of 4th line”, Sum= 152)
- **h.** Using RegEx object check whether given phone number, email address and password is valid or not.
- **i.** Using date module, write a python script to take input as Date of birth (DOB) and current date(CD) and print age of the person.

Program to Perform String Operations (Create, Slice, and Delete)

Description:
This program demonstrates various string operations, including creating a string, slicing it, and deleting a character.

Algorithm:

1. Create a string.
2. Perform slicing to extract a part of the string.
3. Delete a character from the string using replace().

---

b. Program to Check if Two Strings Are Anagrams

Description:
This program checks whether two strings are anagrams, meaning they have the same characters in any order.

Algorithm:

1. Sort both strings.
2. Compare the sorted strings to check if they are equal.

.

---

c. Program to Check if S1 is a Substring of S2

Description:
This program checks if one string (S1) is a substring of another string (S2).

Algorithm:

1. Use the in operator to check if S1 is contained in S2.

---

d. Program to Check if S1 is a Palindrome

Description:
This program checks if a string (S1) is a palindrome, meaning it reads the same forward and backward.

Algorithm:

1. Reverse the string and compare it with the original string.

---

e. Program to Reverse a Given String Using a For Loop

Description:
This program takes a string as input and reverses it using a for loop.

Algorithm:

1. Take a string as input.
2. Use a for loop to reverse the string by appending characters in reverse order.

---

f. Python Script to Find the Sum of All Numbers in a Multi-line String Using Regular Expressions

Description:
This program uses regular expressions to find all numbers in a multi-line string and sum them.

Algorithm:

1. Use the re.findall() method to extract all numbers from the string.
2. Convert the extracted numbers to integers and calculate their sum.

g. Validate Phone Number, Email, and Password Using Regular Expressions

Description:
This program uses regular expressions to validate phone numbers, email addresses, and passwords.

Algorithm:

1. Define regular expressions for phone numbers, emails, and passwords.
2. Use re.match() to check if the input matches the pattern.

h. Python Script to Calculate Age Using Date of Birth and Current Date

Description:
This program calculates a person's age by taking their date of birth and the current date as inputs.

Algorithm:

1. Take the date of birth and current date as input.
2. Calculate the difference between the two dates to determine the age.

Important Viva Questions:

1. **What is a palindrome and how is it checked in Python?**
2. **How does the re.findall() method work in regular expressions?**
3. **What is the difference between re.match() and re.search()?**
4. **What are the basic string manipulation techniques in Python?**
5. **How do you calculate the difference between two dates in Python?**

WEEK 8

Experiment8:

Programs on Lists & Dictionary

a. Write a program to perform the given operations on a list:
   i) Creating the list ii) slicing in the lists iii) Adding Elements in List
   iv) Deleting the list elements

b. Write a program to count the number of vowels in a string(No control flow allowed).

c. Write a program to check if a given key exists in a dictionary or not.

d. Write a program to add a new key-value pair to an existing dictionary.

e. Write a program to take input as String S and print frequency of each character in S using List data structure.

f. Write a program to take input as String S contains characters and special symbols, reverse the String S such that special symbols remains at same position. (Eg. S=”m@d#u” , Output=”u@d#m”).

g. Write a python script to take input as String sentence S and print each word count using dictionary.

h. Write a python script to implement Anonymous function.

i. Write a python script to implement map(), reduce() and filter() functions

a. Program to Perform List Operations (Create, Slice, Add, and Delete)

Description:
This program demonstrates creating a list, slicing it, adding elements, and deleting elements.

Algorithm:

1. Create a list.
2. Perform slicing on the list.
3. Add elements to the list.
4. Delete elements from the list.

---

b. Program to Count the Number of Vowels in a String (No Control Flow)

Description:
This program counts the number of vowels in a string without using control flow statements like if or for.

Algorithm:

1. Use the count() method to count occurrences of each vowel ('a', 'e', 'i', 'o', 'u') in the string.

---

c. Program to Check if a Given Key Exists in a Dictionary

Description:
This program checks whether a given key exists in a dictionary.

Algorithm:

1. Use the in operator to check if the key is in the dictionary.

---

d. Program to Add a New Key-Value Pair to an Existing Dictionary

Description:
This program adds a new key-value pair to an existing dictionary.

Algorithm:

1. Use the assignment operator to add a new key-value pair.

e. Program to Print Frequency of Each Character in a String Using List

Description:
This program calculates the frequency of each character in a string using a list data structure.

Algorithm:

1. Convert the string into a list of characters.
2. Use a dictionary to store the frequency of each character.

---

f. Program to Reverse a String While Keeping Special Symbols in the Same Position

Description:
This program reverses the string while leaving the special symbols in their original positions.

Algorithm:

1. Identify the positions of special symbols.
2. Reverse the non-special characters.
3. Place them back in their original positions.

---

g. Python Script to Count Each Word in a Sentence Using Dictionary

Description:
This program counts the occurrences of each word in a sentence using a dictionary.

Algorithm:

1. Split the sentence into words.
2. Use a dictionary to count the frequency of each word.

---

h. Program to Implement Anonymous Function (Lambda)

Description:
This program demonstrates how to use an anonymous function (lambda) to add two numbers.

Algorithm:

1. Use the lambda keyword to create an anonymous function.
2. Call the lambda function with appropriate arguments.

---

i. Program to Implement map(), reduce(), and filter() Functions

Description:
This program demonstrates the use of map(), reduce(), and filter() functions.

Algorithm:

1. Use map() to apply a function to each item of a list.
2. Use filter() to filter items based on a condition.
3. Use reduce() to reduce a list to a single value.

Viva Questions:

1. **What is a lambda function in Python and how does it work?**

2. **What is the difference between map() and filter() functions in Python?**

3. **What is the role of the reduce() function in Python?**

4. **What is the advantage of using set() in list operations?**

5. **What is the purpose of special symbols in string manipulation?**

WEEK 9

Experiment 9:
Programs on OOPS.

- **a.** Write a Python program to create a person class. Include attributes like name, country and date of birth. Implement a method to determine the person's age.
- **b.** Write a Python program to create a calculator class. Include methods for basic arithmetic operations.
- **c.** Write a Python program to create a class representing a shopping cart. Include methods for adding and removing items, and calculating the total price.
- **d.** Using Python OOPS, create a class, constructor, method,___str__ and___repr__ for
- **e.** Employee
- **f.** Using Python OOPS, create a class, constructor, method, ___str__ and___repr for Student

a. Python Program to Create a Person Class

Description:
This program defines a Person class with attributes like name, country, and date_of_birth. It includes a method to calculate the person's age based on their date of birth.

Algorithm:

1. Define the class Person with attributes: name, country, and date_of_birth.
2. Implement a method calculate_age() that computes the age from the date_of_birth.

b. Python Program to Create a Calculator Class

Description:
This program defines a Calculator class with methods to perform basic arithmetic operations like addition, subtraction, multiplication, and division.

Algorithm:

1. Define the class Calculator with methods for add(), subtract(), multiply(), and divide().
2. Each method takes two parameters and returns the result of the respective operation.

c. Python Program to Create a Shopping Cart Class

Description:
This program defines a ShoppingCart class with methods to add and remove items, and calculate the total price of the cart.

Algorithm:

1. Define the class ShoppingCart with methods:
   - add_item(): Adds an item with its price.
   - remove_item(): Removes an item by its name.
   - calculate_total(): Calculates the total price of items in the cart.

d. Python OOP Program with Class, Constructor, Method, str, and repr for Employee

Description:
This program defines a class Employee with attributes like name, position, and salary. It includes a constructor, a method, and string representations (str and repr).

Algorithm:

1. Define the class Employee with attributes: name, position, and salary.
2. Implement the __str__() method for a user-friendly string representation.
3. Implement the __repr__() method for a more technical string representation.

e. Python OOP Program with Class, Constructor, Method, str, and repr for Student

Description:
This program defines a Student class with attributes like name, age, and marks. It includes a constructor, a method for calculating grades, and string representations (str and repr).

Algorithm:

1. Define the class Student with attributes: name, age, and marks.
2. Implement the __str__() method for a user-friendly string representation.
3. Implement the __repr__() method for a more technical string representation.

Important Viva Questions:

1. **What is the difference between __str__ and __repr__ in Python?**

2. **What are the advantages of using constructors in classes?**

3. **What is the purpose of the self keyword in Python?**

4. **Can you explain method overloading in Python?**

5. **How do you handle exceptions in Python OOPs?**

WEEK 10

Experiment10:

Programs on Exceptions.

a. Write a python program to implement Exceptions hierarchy.
b. Write a program to Catching Specific Exceptions in Python
c. Python program to try with else clause.
d. Write a Python program to handle a ZeroDivisionError exception when dividing a number by zero.
e. Create a user defined Exception named “FundsLessException” and raise the exception when there are no enough funds in the bank account.

a. Python Program to Implement Exceptions Hierarchy

Description:
This program demonstrates the exception hierarchy by creating multiple custom exception classes. It shows how exceptions can inherit from other exceptions.

Algorithm:

1. Create a base exception class BaseException.
2. Create other exception classes like CustomException1 and CustomException2 that inherit from BaseException.
3. Raise and catch exceptions using a try-except block.

c. Python Program to Try with Else Clause

Description:
This program demonstrates the use of the else clause in a try-except block. The code inside the else block runs if no exceptions are raised.

Algorithm:

1. Use a try block to attempt an operation.
2. Use the else block to execute code if no exception occurs.

d. Python Program to Handle ZeroDivisionError Exception

Description:
This program specifically handles the ZeroDivisionError exception when trying to divide a number by zero.

Algorithm:

1. Prompt the user for input and attempt to divide two numbers.

2. If a division by zero occurs, catch the ZeroDivisionError.

---

e. Create a User Defined Exception Named "FundsLessException"

Description:
This program defines a custom exception called FundsLessException to handle cases where a bank account has insufficient funds.

Algorithm:

1. Define a custom exception FundsLessException.
2. Create a function that checks the account balance and raises FundsLessException if funds are insufficient.

Important Viva Questions:

1. **What is the difference between try-except and try-except-finally in Python?**
2. **How do you create a custom exception in Python?**
3. **What is the purpose of the else block in a try-except structure?**
4. **How does exception hierarchy work in Python?**
5. **What is the use of the raise keyword in Python?**

WEEK 11

Experiment11:

Programs on Data Analysis

- **a.** Python Program to demonstrate NumPy arrays creation using array()function.
- **b.** Python script to load data sets.
- **c.** Write a python script to create a data frame.
- **d.** Python program to demonstrate use of ndim, shape,size,dtype.
- **e.** Using NumPy, implement different matrix operations in python.
- **f.** Using pandas, read the data from anytext files.
- **g.** Python program to find min, max, sum, cumulative sum of array
- **h.** Create a dictionary with at least five keys and each key represent value as a list where this list contains at least ten values and convert this dictionary as a pandas data frame and explore the data through the data frame as follows: Apply head() function to the pandas data frame
- **i.** Perform various data selection operations on Data Frame

a. Python Program to Demonstrate NumPy Arrays Creation Using array() Function

Description:
This program demonstrates how to create NumPy arrays using the array() function. The array() function takes a list or a tuple as input to create an array.

Algorithm:

1. Import the NumPy library.
2. Create a NumPy array using the np.array() function.
3. Print the created array.

b. Python Script to Load Datasets

Description:
This program demonstrates how to load datasets from a file (e.g., CSV) using the pandas library. This is useful for data analysis tasks.

Algorithm:

1. Import the pandas library.
2. Use the pandas.read_csv() function to load a dataset from a CSV file.
3. Display the dataset.

c. Write a Python Script to Create a DataFrame

Description:
This program demonstrates how to create a DataFrame using the pandas library. A DataFrame is a 2-dimensional labeled data structure.

Algorithm:

1. Import the pandas library.
2. Create a dictionary with sample data.
3. Convert the dictionary to a DataFrame using pd.DataFrame().

d. Python Program to Demonstrate Use of ndim, shape, size, dtype

Description:
This program demonstrates the use of NumPy array attributes: ndim, shape, size, and dtype.

Algorithm:

1. Create a NumPy array.
2. Print the number of dimensions (ndim), shape (shape), size (size), and data type (dtype) of the array.

e. Using NumPy, Implement Different Matrix Operations in Python

Description:
This program demonstrates various matrix operations (addition, multiplication, transpose, etc.) using the NumPy library.

Algorithm:

1. Create two NumPy arrays (matrices).
2. Perform matrix operations like addition, multiplication, and transpose.

f. Using Pandas, Read the Data from Any Text File

Description:
This program demonstrates how to use the pandas library to read data from a text file. This can be useful for data analysis tasks where the data is stored in text format.

Algorithm:

1. Import the pandas library.
2. Use pd.read_csv() with a file path to load data from a text file (assuming the file is comma-separated).

g. Python Program to Find Min, Max, Sum, Cumulative Sum of Array

Description:
This program demonstrates how to compute the minimum, maximum, sum, and cumulative sum of elements in a NumPy array using built-in functions.

Algorithm:

1. Import the NumPy library.
2. Create a NumPy array.
3. Use np.min(), np.max(), np.sum(), and np.cumsum() to find the required statistics.
4. Print the results.

h. Create a Dictionary with at Least Five Keys, and Each Key Represents a List of Ten Values. Convert this Dictionary to a Pandas DataFrame and Explore the Data through the DataFrame

Description:
This program demonstrates how to create a dictionary with lists as values, convert it to a Pandas DataFrame, and use the head() function to explore the data.

Algorithm:

1. Create a dictionary with keys and values (where each value is a list).
2. Convert the dictionary to a Pandas DataFrame.
3. Use the head() function to display the first few rows of the DataFrame.

i. Perform Various Data Selection Operations on DataFrame

Description:
This program demonstrates how to perform various data selection operations on a Pandas DataFrame, such as selecting rows, columns, and performing conditional selection.

Algorithm:

1. Create a DataFrame.
2. Select specific columns.
3. Filter rows based on conditions.
4. Select specific rows using loc[] and iloc[].

Viva Questions:

1. **What is the difference between loc[] and iloc[] in Pandas?**
2. **What is the use of the head() function in Pandas?**
3. **How do you perform conditional filtering on a Pandas DataFrame?**
4. **What is the difference between min() and cumsum() in NumPy?**
5. **Explain how to convert a dictionary to a Pandas DataFrame.**

WEEK 12

Experiment12:

Programs on Plotting

- **a.** Create a line plot for a list of values over a period of time. Label the x-axis as "Time" and the y-axis as "Value".
- **b.** Create a bar chart that shows the frequency of categories in a dataset. Use different colors for each bar.
- **c.** Create a histogram to visualize the distribution of a numerical dataset. Customize the number of bins and add a title.
- **d.** Create a scatter plot to show the relationship between two numerical variables. Add a trend line to the scatter plot.
- **e.** Create a pie chart to show the proportion of different categories in a dataset. Add labels and percentages to each slice.

a. Create a Line Plot for a List of Values Over a Period of Time. Label the X-axis as "Time" and the Y-axis as "Value".

Description:
This program demonstrates how to create a line plot using matplotlib. It will show a list of values over a specified time period.

Algorithm:

1. Import the matplotlib.pyplot library.
2. Define the time and value lists.
3. Use plt.plot() to create the line plot.
4. Label the axes using plt.xlabel() and plt.ylabel().
5. Display the plot using plt.show().

b. Create a Bar Chart that Shows the Frequency of Categories in a Dataset. Use Different Colors for Each Bar.

Description:
This program demonstrates how to create a bar chart using matplotlib to visualize the frequency of categories in a dataset.

Algorithm:

1. Import the matplotlib.pyplot library.
2. Define the categories and their corresponding frequencies.
3. Use plt.bar() to create the bar chart with different colors for each bar.
4. Display the chart using plt.show().

c. Create a Histogram to Visualize the Distribution of a Numerical Dataset. Customize the Number of Bins and Add a Title.

Description:
This program creates a histogram to visualize the distribution of a numerical dataset using matplotlib.

Algorithm:

1. Import the matplotlib.pyplot library.
2. Define the numerical data for the histogram.
3. Use plt.hist() to create the histogram and set the number of bins.
4. Add a title to the plot using plt.title().
5. Display the plot using plt.show().

the Relationship Between Two Numerical Variables. Add a Trend Line to the Scatter Plot.

Description:
This program creates a scatter plot and adds a trend line using matplotlib.

Algorithm:

1. Import matplotlib.pyplot and numpy for generating data and fitting a line.
2. Define the two numerical variables (x and y).
3. Use plt.scatter() to create the scatter plot.
4. Fit a trend line using numpy.polyfit() and plot it.
5. Display the plot using plt.show().

e. Create a Pie Chart to Show the Proportion of Different Categories in a Dataset. Add Labels and Percentages to Each Slice.

Description:
This program creates a pie chart using matplotlib to show the proportions of different categories in a dataset.

Algorithm:

1. Import the matplotlib.pyplot library.
2. Define the categories and their corresponding values.
3. Use plt.pie() to create the pie chart, with labels and percentages.
4. Display the plot using plt.show().

Viva Questions:

1. **What is the purpose of plt.xlabel() and plt.ylabel() in plotting?**
2. **What does plt.show() do in matplotlib?**
3. **How do you change the colors of bars in a bar chart in matplotlib?**
4. **What is the purpose of autopct='%1.1f%%' in a pie chart?**
5. **How do you add a trend line to a scatter plot?**